Austin Animal Center: An Analysis

By Project A-Team

Our objective was to analyze the intakes and outcomes of animals in the shelter to observe trends by zip code and predict adoptability by characteristics.

If analysis shows that certain traits make an animal more adoptable, the AAC can focus its efforts on those animals to improve availability. Conversely, if the AAC can focus education efforts in areas that have high intake volumes.

The Players

Role	Responsibility	Name
Square 	Github	Corinne Bean
Circle 	Machine Learning	Shruti Ramana
Triangle 	Database	Sharon Dieckert
Exes 	Project Management	Nick Foley & Ashley Rock

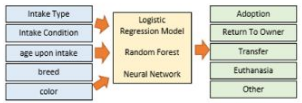
Project Workflow - Database & Machine Learning

Data Sources:
Austin Animal Center Intake Data
Austin Animal Center Outcome Data
Texas A&M GeoServices
US Cenus Bureau

PostgreSQL13
Austin Animal Center Intake Data
Austin Animal Center Outcome Data
Texas A&M GeoServices Lat and Long
US Cenus Bureau

Output Files
acc_intake_outcome.csv
acc_intake_available

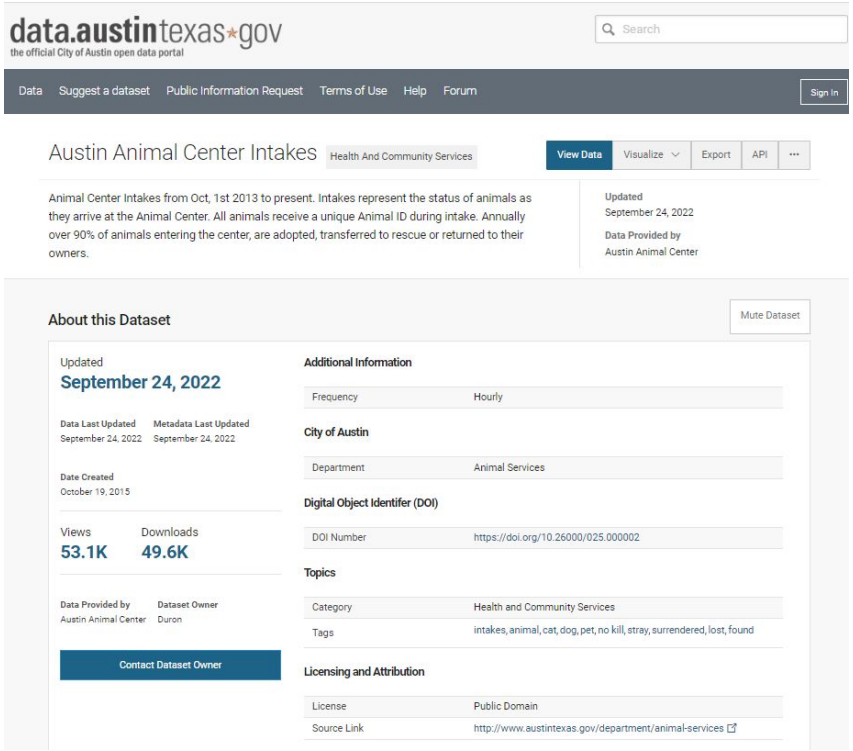
Jupyter Notebook
Apply analysis and
Filter for ML

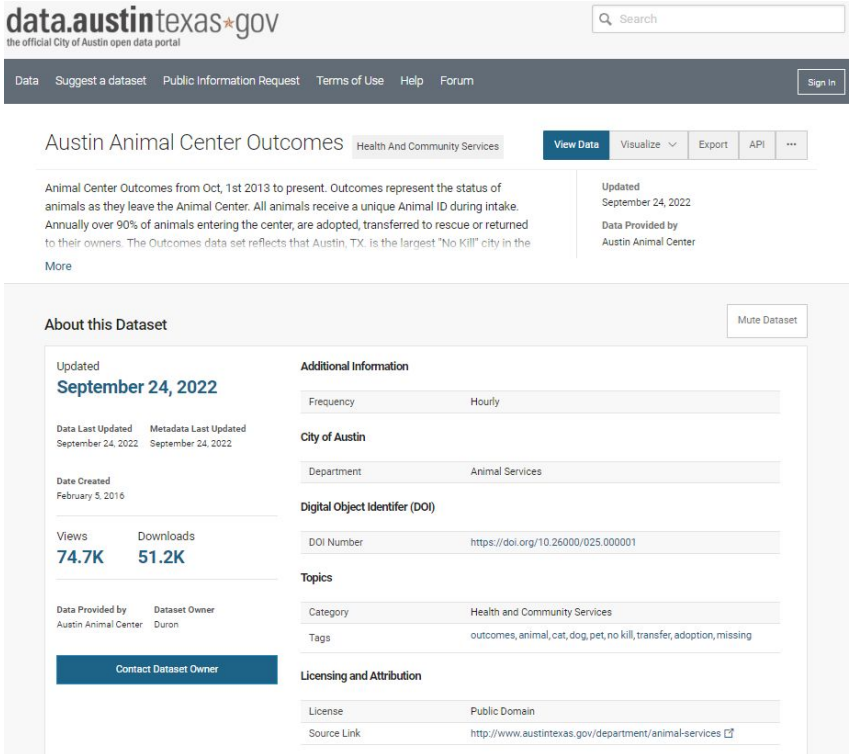


Machine Learning
Logistic Regression
Random Forest
Neural Networks

Tableua
Data Visualization and Story Telling

Data Sources



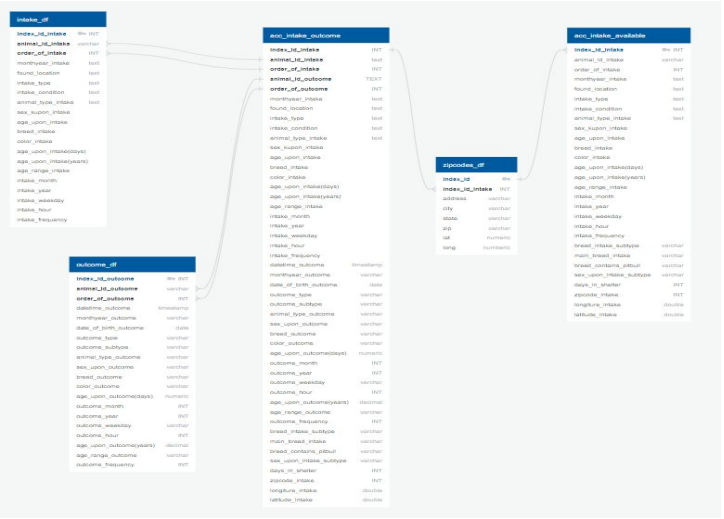


Github

Corinne, this is where you do your thing! Maybe a screenshot of GH? Include some speaker notes.

Database

Sharon, quick overview of the DB? I've put the DB slides below this, but think it should be consolidated and add speaker notes



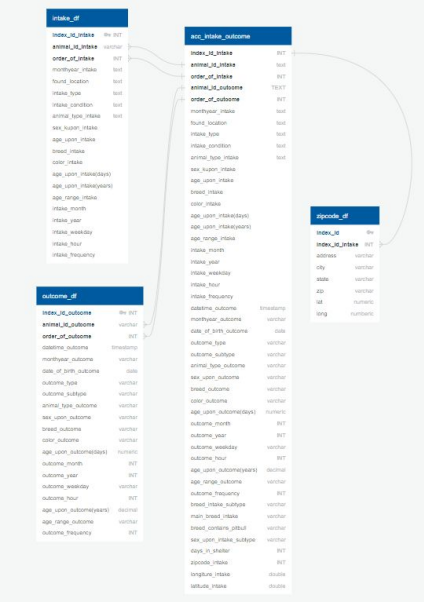
Database Integration

For this project, we utilized PostgresSQL and fully integrated the database into our project.

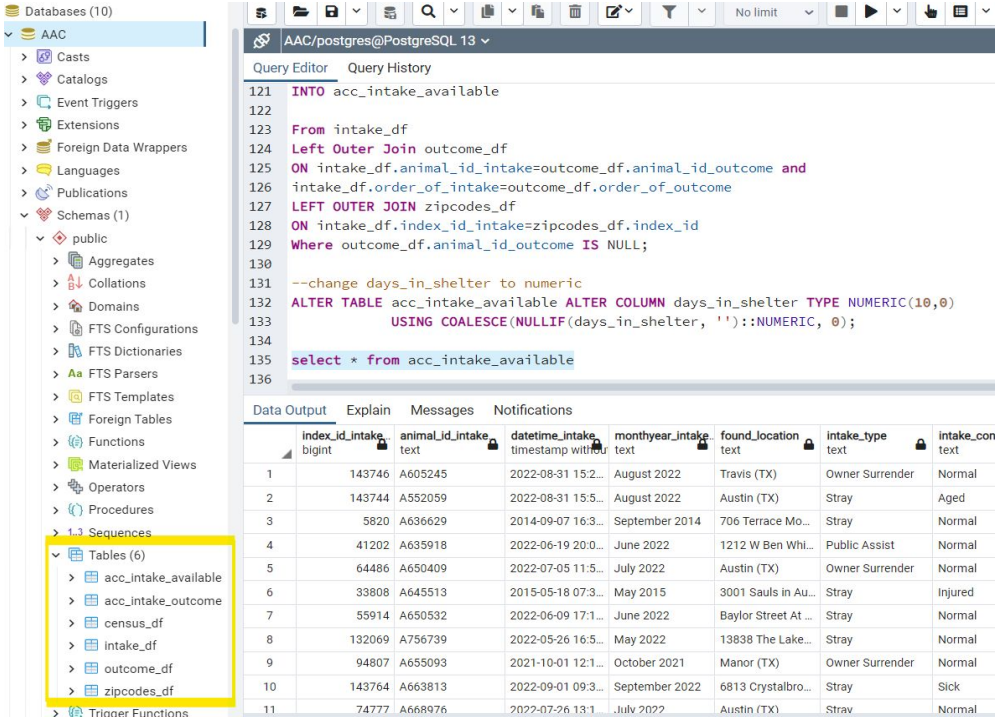
- Database stores static data for use during the project
  - *Database stores multiple data tables used for compiling the final dataset.*
- Database interfaces with the project in some format (e.g., scraping updates the database)
  - *Database interfaces with the project using Jupyter Notebook Pandas and sqlalchemy to export the dfs into the Postgres AAC database and create the tables and to import the final sql table back to Jupyter Notebooks to use for the machine learning.*
- Includes at least two tables (or collections, if using MongoDB)
  - *The AAC database includes three tables used for the final dataset.*
- Includes at least one join using the database language (not including any joins in Pandas)
  - *SQL is used to join the three tables together and perform data manipulation on number, character and date columns.*
- Includes at least one connection string (using SQLAlchemy or PyMongo)
  - *Three connection strings using SQLAlchemy export the data into the Postgres database and two connection strings import the final dataset into the Jupyter Notebook machine learning scripts.*

Workflow of Project: Database

Database ERD



Postgres Database Stores Data Tables



Database Interfaces With AAC_ML_model_Adopted(Success_Failure) - Dog/Cat

In [2]:
```
import pandas as pd
import sklearn as skl
from sklearn.preprocessing import OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
```

In [3]:
```
from sqlalchemy import create_engine
from config import db_password
```

In [4]:
```
# Create a connection with the database in postgres
db_string = f"postgresql://postgres:{db_password}@127.0.0.1:5432/AAC"
```

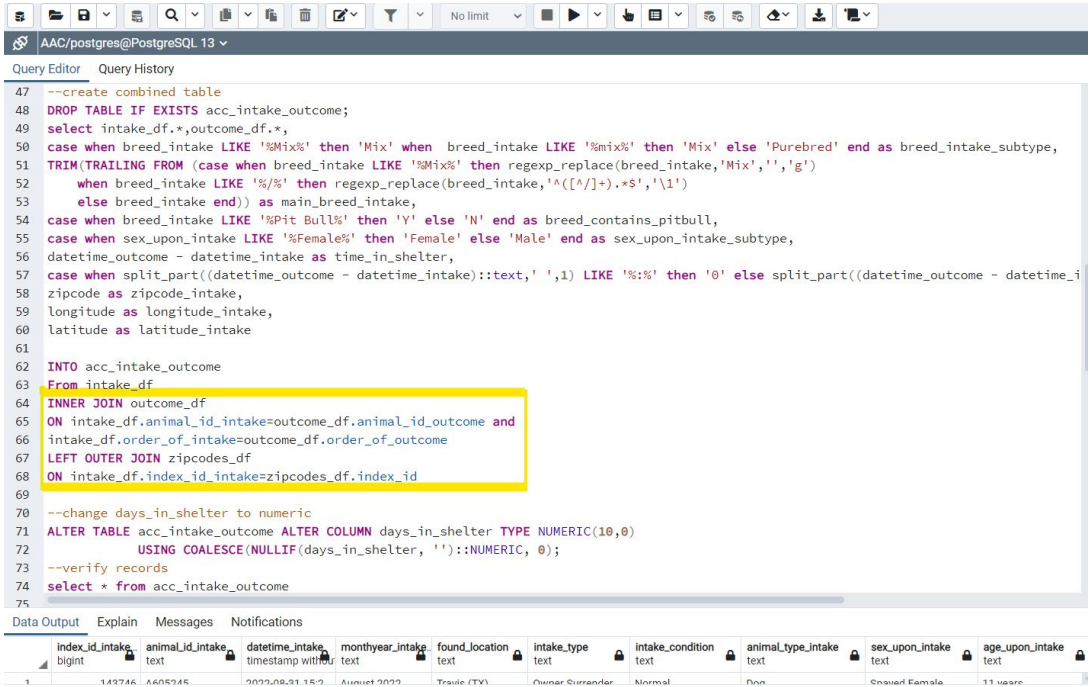
In [5]:
```
engine = create_engine(db_string)
```

In [6]:
```
# read the table from the database
df = pd.read_sql_table("acc_intake_outcome",engine)
df.head()
```

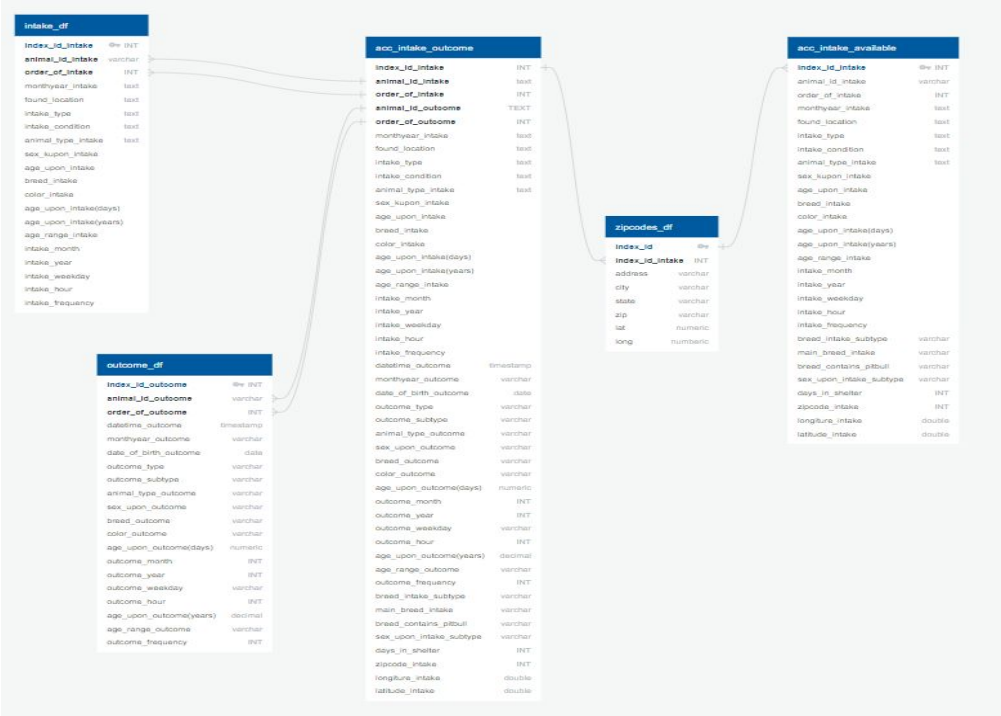
Out[6]:

	index_id_intake	animal_id_intake	datetime_intake	monthyear_intake	found_location	intake_type	intake_condition	animal_type_intake	sex_upon_intake
0	70641	A178569	2014-03-17 09:45:00	March 2014	Austin (TX)	Public Assist	Normal	Dog	Neutered Male
1	944	A287017	2015-08-16 12:19:00	August 2015	6620 Deatonhill Dr in Austin (TX)	Stray	Normal	Dog	Spayed Female
2	69127	A293383	2018-03-18 18:17:00	March 2018	6005 Walnut Hills in Austin (TX)	Stray	Sick	Cat	Neutered Male

Database Includes Joins. Refer to full SQL @ /Database/aac.sql



ERD


intake_df
index_id_intake PK INT
animal_id_intake varchar
order_of_intake INT
monthyear_intake text
found_location text
intake_type text
intake_condition text
animal_type_intake text
sex_kupon_intake
age_upon_intake
breed_intake
color_intake
age_upon_intake(days)
age_upon_intake(years)
age_range_intake
intake_month
intake_year
intake_weekday
intake_hour
intake_frequency
outcome_df
index_id_outcome PK INT
animal_id_outcome varchar
order_of_outcome INT
datetime_outcome timestamp
monthyear_outcome varchar
date_of_birth_outcome date
outcome_type varchar
outcome_subtype varchar
animal_type_outcome varchar
sex_upon_outcome varchar
breed_outcome varchar
color_outcome varchar
age_upon_outcome(days) numeric
outcome_month INT
outcome_year INT
outcome_weekday varchar
outcome_hour INT
age_upon_outcome(years) decimal
age_range_outcome varchar
outcome_frequency INT
acc_intake_outcome
index_id_intake INT
animal_id_intake text
order_of_intake INT
animal_id_outcome TEXT
order_of_outcome INT
monthyear_intake text
found_location text
intake_type text
intake_condition text
animal_type_intake text
sex_kupon_intake
age_upon_intake
breed_intake
color_intake
age_upon_intake(days)
age_upon_intake(years)
age_range_intake
intake_month
intake_year
intake_weekday
intake_hour
intake_frequency
datetime_outcome timestamp
monthyear_outcome varchar
date_of_birth_outcome date
outcome_type varchar
outcome_subtype varchar
animal_type_outcome varchar
sex_upon_outcome varchar
breed_outcome varchar
color_outcome varchar
age_upon_outcome(days) numeric
outcome_month INT
outcome_year INT
outcome_weekday varchar
outcome_hour INT
age_upon_outcome(years) decimal
age_range_outcome varchar
outcome_frequency INT
bread_intake_subtype varchar
main_bread_intake varchar
bread_contains_pitbull varchar
sex_upon_intake_subtype varchar
days_in_shelter INT
zipcode_intake INT
longitude_intake double
latitude_intake double
zipcodes_df
index_id PK
index_id_intake INT
address varchar
city varchar
state varchar
zip varchar
lat numeric
long numeric
acc_intake_available
index_id_intake PK INT
animal_id_intake varchar
order_of_intake INT
monthyear_intake text
found_location text
intake_type text
intake_condition text
animal_type_intake text
sex_kupon_intake
age_upon_intake
breed_intake
color_intake
age_upon_intake(days)
age_upon_intake(years)
age_range_intake
intake_month
intake_year
intake_weekday
intake_hour
intake_frequency
bread_intake_subtype varchar
main_bread_intake varchar
bread_contains_pitbull varchar
sex_upon_intake_subtype varchar
days_in_shelter INT
zipcode_intake INT
longitude_intake double
latitude_intake double

Machine Learning

Shruti does her thing. I would suggest a simple overview “We wanted to see if we could predict the probability of an animal being adopted based on features such as breed, color, age, time in shelter, and health.

To create a binary option, we declared a “Successful” outcome as one where the animal was adopted (But we removed RTO from the success list, right? My memory is a sieve). Failed outcomes include animals that remain in the shelter, were transferred to partner shelters, were euthanized or died while in AAC custody (AND RTO?).

I created logistic regression and random forest models that took in these features and and predicted the probability that an animal would be adopted or not. The results were this for cats and this for dogs.

Some of the issues that arose were low recall rates because of the high rates of success compared with failures at the shelter. As a result I modified the models by adjusting our success and fail metrics to make the sample sizes more even (Or whatever you did!).

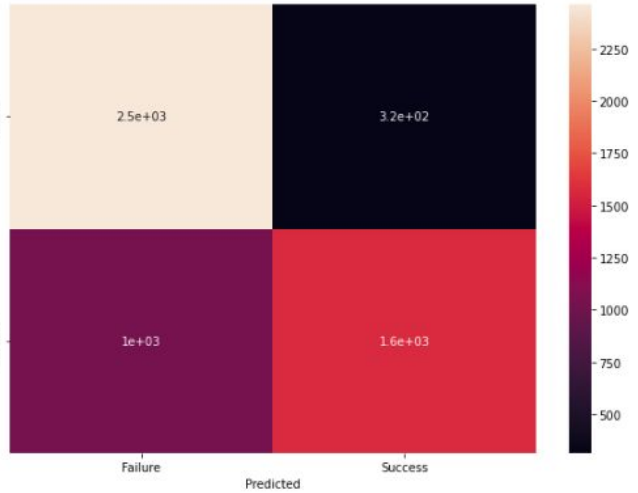
Logistic Regression Results - Dogs

Confusion Matrix for Dogs

	Failure	Success
Failure	4212	1944
Success	1943	3649

Accuracy Score : 0.6691351719441607
Classification Report

	precision	recall	f1-score	support
0	0.68	0.68	0.68	6156
1	0.65	0.65	0.65	5592
accuracy			0.67	11748
macro avg	0.67	0.67	0.67	11748
weighted avg	0.67	0.67	0.67	11748



Logistic Regression Results – Cats

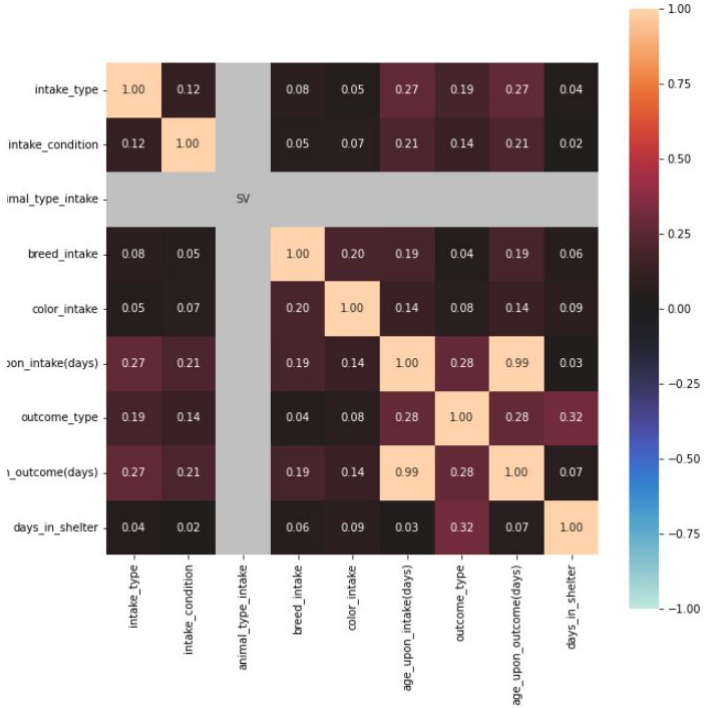
Confusion Matrix for Cats

```
Failure    11106
Success    10399
```

```
report = classification_report(y_test, y_pred)
print(report)
```

	precision	recall	f1-score	support
0	0.70	0.89	0.78	2777
1	0.84	0.58	0.69	2600
accuracy			0.74	5377
macro avg	0.77	0.74	0.74	5377
weighted avg	0.76	0.74	0.74	5377

Categorical Correlation Heat Map for Random Forest Model



2. intake_condition is categorized into three main categories - Normal , Aged & Other - and then they are hot encoded and merged into the main dataframe df_dog_ML

```
intake_condition_normal = ['Normal','Behavior']
intake_condition_aged = ['Aged']
intake_condtion_other = ['Injured','Sick','Nursing','Neonatal','Other','Medical','Feral','Pregnant','Med Urgent']
```

3. outcome_type is categorized into two main categories - Success & Failure - and then they are hot encoded and merged into the main dataframe df_dog_ML.


```
intake_condition_normal = ['Normal','Behavior']
intake_condition_aged = ['Aged']
intake_condtion_other = ['Injured','Sick','Nursing','Neonatal','Other','Medical','Feral','Pregnant','Med Urgent']
```

3. outcome_type is categorized into two main categories - Success & Failure - and then they are hot encoded and merged into the main dataframe df_dog_ML.

```
: # # Determine which values to replace
  # replace_intake_condition = list(intake_type[intake_type < 465].index)

  other_outcome_type_list = ['Transfer','Euthanasia','Died','Disposal','Missing']
  success_outcome_list = ['Adoption','Return to Owner','Rto-Adopt']


  # # Replace in DataFrame
  for outcome in other_outcome_type_list:
      df_dog_ML.outcome_type = df_dog_ML.outcome_type.replace(outcome,"Failure")

  for outcome in success_outcome_list:
      df_dog_ML.outcome_type = df_dog_ML.outcome_type.replace(outcome,"Success")


  # # Check to make sure binning was successful
  df_dog_ML.outcome_type.value_counts()
```

Then we used the function get_dummies on df_dog_ML["outcome_type"] and then merging into the dataframe df_dog_ML.

✓ Description of how data was split into training and testing sets

The data is split into X and y. Where below are the features -
X = The hot encoded values of the following features

- ‘age_upon_intake(days)’
- ’age_upon_outcome(days)’
- ’days_in_shelter’
- ’intake_condition’
- ‘color_intake’
- ‘breed_intake’
- ’intake_type’

Y = ‘outcome_type’ encoded for Success and Failure.

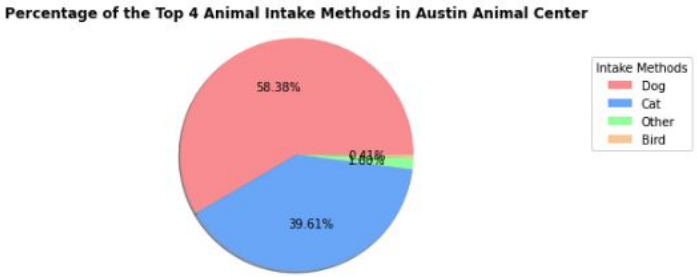
```
# Seperate the features X from the target Y
y = df_dog_ML.Success
columns=["Success","Failure"]
X = df_dog_ML.drop(columns=columns)


# Split training/test datasets
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=42, stratify=y)
```

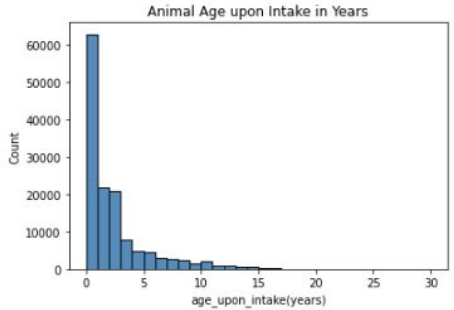
✓ Explanation of model choice, including limitations and benefits.

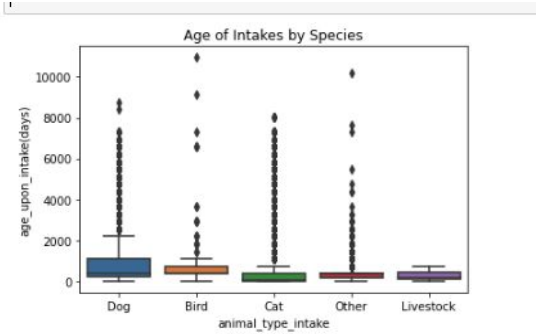
We used Logistic Regression and Random Forest Classification models to analyze the data.

Logistic Regression - is performed when we are expecting a Binary Outcome - Here we are running the ML model to determine if Dog / Cat will have success or Failure as outcome for given categories or features in consideration.
Random Forest Classification - This model produces good predictions, and is capable to handle large datasets efficiently. This model helps in producing higher level of accuracy. Below is the Confusion Matrix for Dogs and Cats.

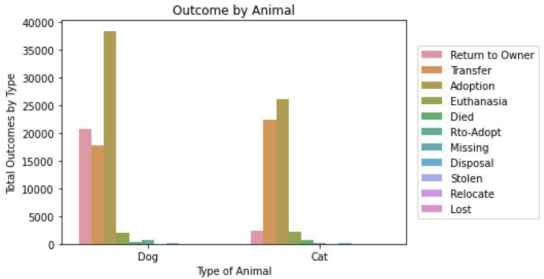
Exploratory Data Analysis - Intake Data

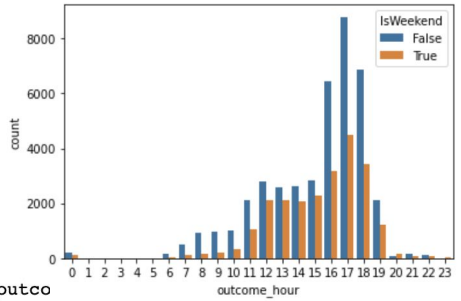




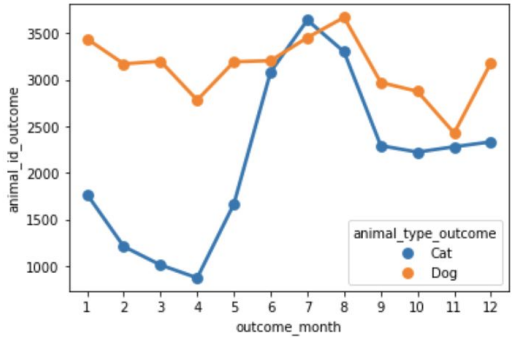


Exploratory Data Analysis - Outcome Data

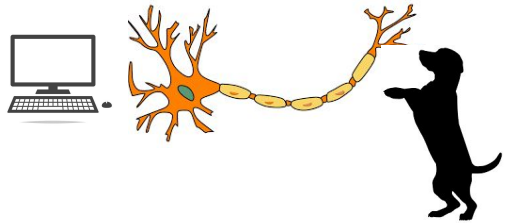





```
<AxesSubplot:xlabel='outcome_month', ylabel='animal_id_outco
```



Bonus Models



Neural Network Results

```
# Evaluate the model using the test data
model_loss, model_accuracy = nn.evaluate(X_test_scaled,y_test,verbose=2)
print(f"Loss: {model_loss}, Accuracy: {model_accuracy}")
```

```
64/64 - 0s - loss: 0.7019 - accuracy: 0.6532 - 203ms/epoch - 3ms/step
Loss: 0.7019389867782593, Accuracy: 0.6532416343688965
```

Logistic Regression Results

	Fail	Success
Fail	53	621
Success	33	1,329

```
report = classification_report(y_test, y_pred)
print(report)
```

	precision	recall	f1-score	support
0.0	0.62	0.08	0.14	674
1.0	0.68	0.98	0.80	1362
accuracy			0.68	2036
macro avg	0.65	0.53	0.47	2036
weighted avg	0.66	0.68	0.58	2036

Bonus Model Pt. 2

Random Forest Results

	Fail	Success
Fail	246	428
Success	144	1,218

Accuracy Score : 0.7190569744597249
Classification Report

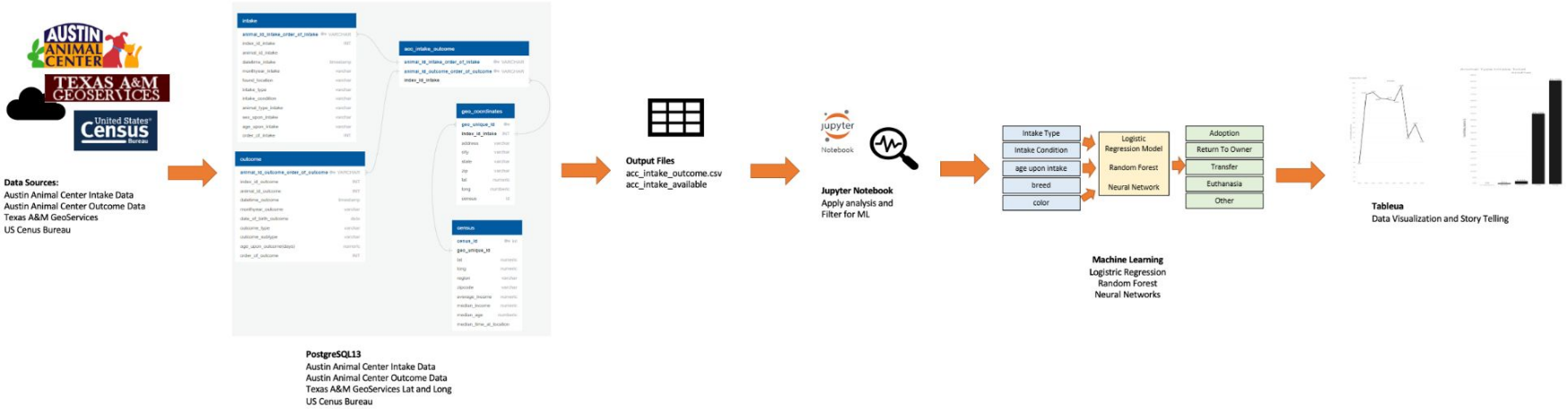
	precision	recall	f1-score	support
0.0	0.63	0.36	0.46	674
1.0	0.74	0.89	0.81	1362
accuracy			0.72	2036
macro avg	0.69	0.63	0.64	2036
weighted avg	0.70	0.72	0.69	2036

SVM Results ... tsk tsk. Not a good model.

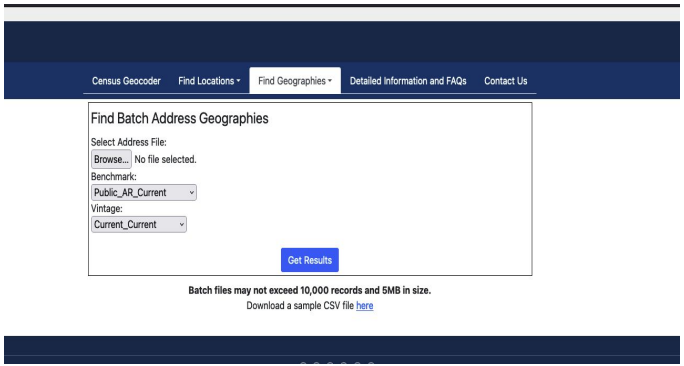
	Fail	Success
Fail	0	674
Success	0	1,362

	precision	recall	f1-score	support
0.0	0.00	0.00	0.00	674
1.0	0.67	1.00	0.80	1362
accuracy			0.67	2036
macro avg	0.33	0.50	0.40	2036
weighted avg	0.45	0.67	0.54	2036

Data Analysis



Geocoding and Demographics Data



https://geocoding.geo.census.gov/geocoder/geographies/addressbatch?form

```
demographic_df.head()
```

	zipcode	major_city	county	state	population	population_density	land_area_in_sqmi	housing_units	median_household_income	population_by_age	population
0	78724	Austin	Travis County	TX	21696	889.0	24.40	6138	38479	[{'key': 'Male', 'values': [{'x': 0, 'y': 1163...	[{'key': 'D [{'x':
1	78660	Pflugerville	Travis County	TX	68789	1519.0	45.30	23950	76007	[{'key': 'Male', 'values': [{'x': 0, 'y': 2928...	[{'key': 'D [{'x':
2	78747	Austin	Travis County	TX	14808	623.0	23.78	5491	61599	[{'key': 'Male', 'values': [{'x': 0, 'y': 599}...	[{'key': 'D [{'x':
3	78732	Austin	Travis County	TX	14060	1061.0	13.25	5033	131216	[{'key': 'Male', 'values': [{'x': 0, 'y': 771}...	[{'key': 'D [{'x':
4	78728	Austin	Travis County	TX	20299	2503.0	8.11	10240	48612	[{'key': 'Male', 'values': [{'x': 0, 'y': 883}...	[{'key': 'D [{'x':

```
demographic_df.to_csv("demographic_details.csv")
```

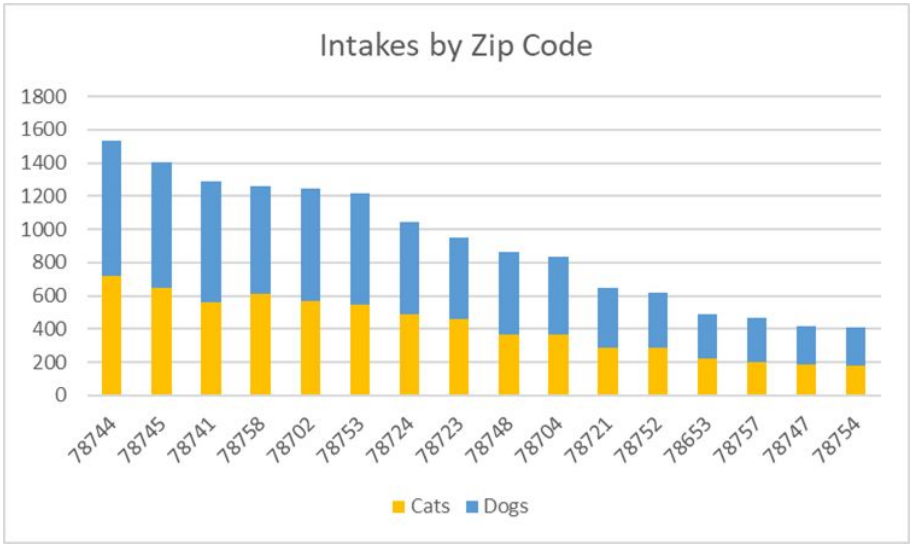
population_by_age		population_by_gender	population_by_race	educational_attainment_for_population_25_and_over	school_enrollment_age_3_to_17
Female, Male, Total		0,1	0,1,2,3,4,5,6	0,1,2,3,4,5,6	0,1,2
Under 5 years	0	Male : values	White	Less Than High School Diploma	Enrolled in Public School
5 to 9 years	1	Female : values	Black Or African American	High School Graduate	Enrolled in Private School
10 to 14 years	2		American Indian Or Alaskan Native	Associate's Degree	Not Enrolled In school
15 to 19 years	3		Asian	Bachelor's Degree	
20 to 24 years	4		Native Hawaiian & Other Pacific Islander	Master's Degree	
25 to 29 years	5		Other Race	Professional School Degree	
30 to 34 years	6		Two Or More Races	Doctorate Degree	
35 to 39 years	7				
40 to 44 years	8				
45 to 49 years	9				
50 to 54 years	10				
55 to 59 years	11				
60 to 64 years	12				
65 to 69 years	13				
70 to 74 years	14				
75 to 79 years	15				
80 to 84 years	16				
85 years and over	17				

Data Exploration:

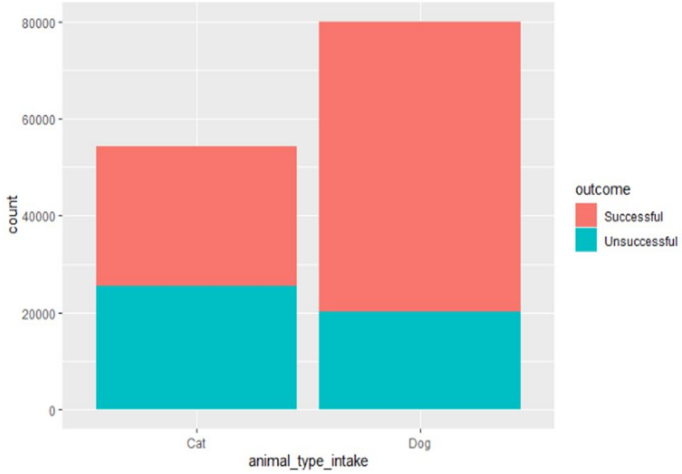
Where are the animals coming from?

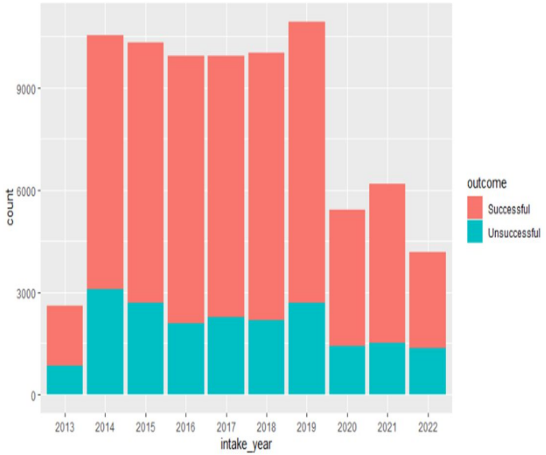
We will use a layered map in our dashboard to easily show common areas for strays.

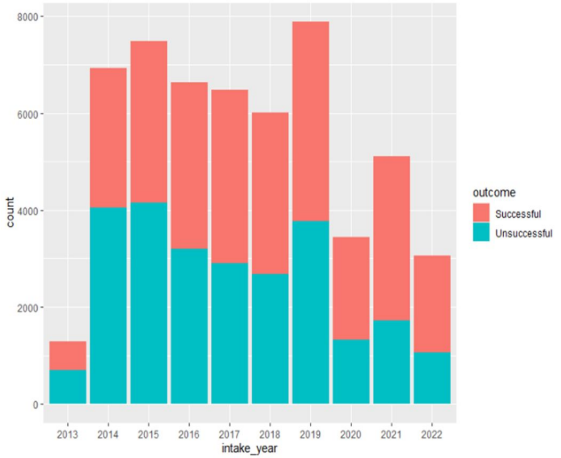
What is the proportion of animals that are brought to the AAC?



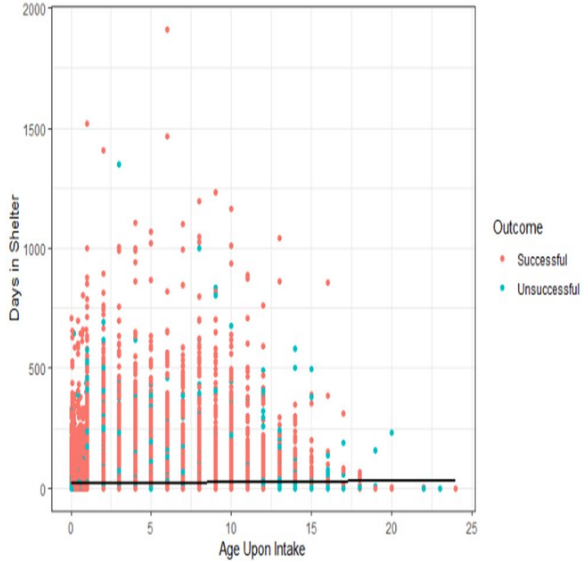
Overview of Outcomes (Sharon could we get these graphs in the without the legend? We just need one legend for all the graphs… this would be a great one to have interactive to show all outcomes (we do need to define how successful was defined… did you include RTO and adopted? Vs unsuccessful, although we also changed the meaning of unsuccessful to include animals in the shelter but that's ok)

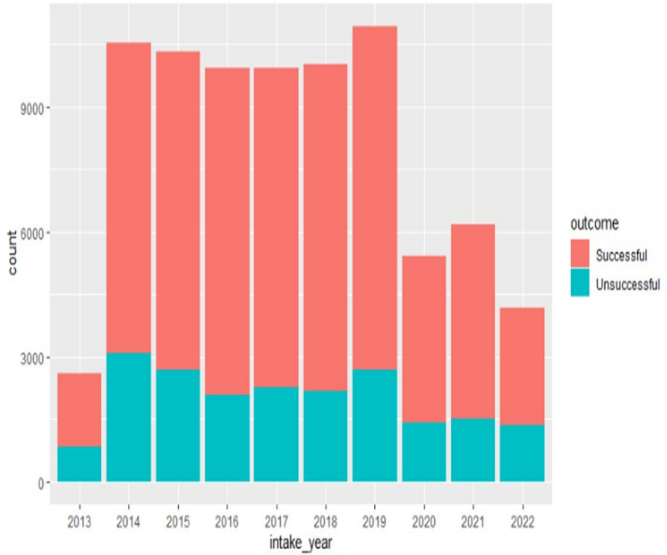




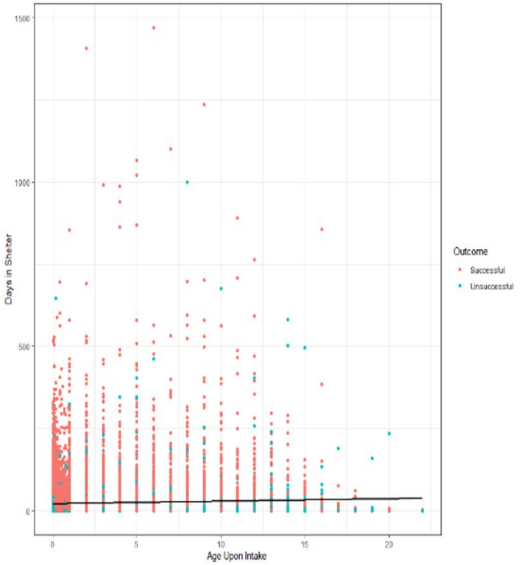


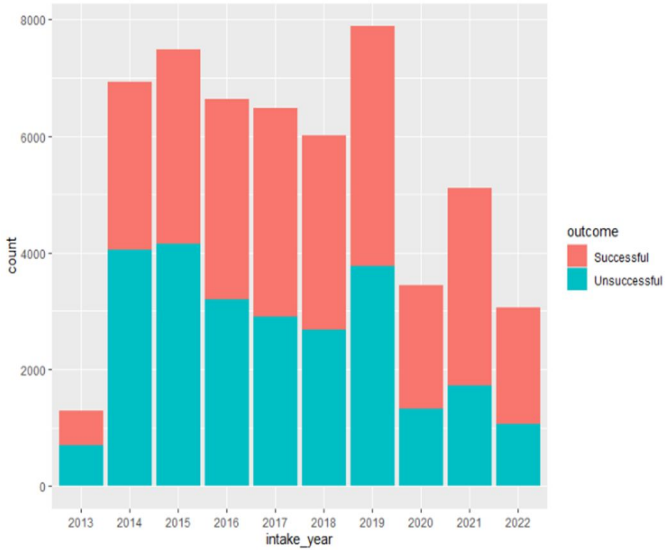
Dogs Categorized As Successful/Unsuccessful Outcome





Cats Categorized As Successful/Unsuccessful Outcome





Cats vs Dogs

Pie chart of outcomes for cats and dogs, chart showing most common breeds of dogs on intake, chart showing most common cat on intake

By the Numbers

Average time in shelter, average age of dogs on intake/outcome, average age of cats on intake/outcome.

Perhaps box and whisker plot on average time in shelter and simple chart of the ages?

A heat map that shows most popular day of the week to adopt?

Demographics on Intake Areas

Stacked bar chart based on median income and population density? Line graph on intakes vs income? Income vs population? I don’t think there is a linear relationship but perhaps? Just trying to vary up the types of graphs from bar chart.

Recommendations & Improvements

Recommendations

- Get address information for outcomes to see if people from certain areas adopt more frequently

Conclusion/Proposals

1). Drawing conclusions from our questions we solve with our data analysis/storytelling.

2). Propose ideas on how AAC could improve adoptability by targeting animals that are most adoptable to get them out of the shelter faster.

3). Propose new adoption programs that can improve adoption rates of older animals, lower cost upfront, extra supplies, and more.

4). Show pers how many pets are available in a local shelter in Austin.

The Nuts and Bolts of our Analysis

Technologies	Languages	Tools
GitHub	Python	Google Slides
Tableau	R	
Jupyter Notebook	SQL	

Workflow of Project: Machine Learning Model

The following is a provisional machine learning model which was used during the planning